UNITED STATES PATENT APPLICATION

FOR

METHOD AND SYSTEM FOR PRESERVING AN ORIGINAL TABLE SCHEMA

Inventor(s):

Craig A. FRISKE
Regina J. LIU
James A. RUDDY
James Z. TENG
Julie A. WATTS

Sawyer Law Group LLP
2465 E. Bayshore Road
Suite 406
Palo Alto, CA 94303

METHOD AND SYSTEM FOR PRESERVING AN ORIGINAL TABLE SCHEMA

FIELD OF THE INVENTION

The present invention relates to database management systems and more particularly to a method and system for preserving an original table schema in a database system that supports dynamic table schema changes.

BACKGROUND OF THE INVENTION

A schema provides a definition of a database table. The schema defines the structure and the type of contents that each data element within the structure can contain. For example, the schema for a structure that includes a table defines the size of a column in the table and the type of data in the column. The schema is generally stored in the database management system's (DBMS's) system database or catalog.

Modern database management systems, such as the DB2™ system developed by International Business Machines of Armonk, New York, support dynamic table schema changes by introducing self-describing table rows. In this system, each table is associated with a metadata table that contains information related to the table and each row's existing schema. The metadata is stored with the table, and therefore, each row is self-describing, i.e., reference to any source besides the table is not required. When the schema is altered, the latest, i.e., post alteration, table definition is recorded in the catalog. For any new row inserted into the table after the schema alteration, the corresponding metadata reflects the latest table definition. An existing row's metadata remains unchanged until a request to update that row is executed. At that time, the existing row's metadata is updated to reflect

the latest table definition. Otherwise, the existing row's metadata reflects the pre-alteration table definition.

The above described dynamic schema alteration function performs well for its intended purpose and it is desirable to implement such functionality in a database system that includes tables that are not self-describing, i.e., table definitions are not stored with the table data. After the dynamic schema alteration function is invoked for the first time in such a database system, i.e., when the schema is altered for the first time, and when a first update to the table is performed, the DBMS appends both the old, i.e., original, and the new table definitions to the metadata table. Note that the table definitions are appended at this time for purposes of optimization. Those skilled in the art recognize that the table definitions can be appended at a different time, e.g., when the schema is first changed. Thus, after the first table update, each row is self-describing. As new rows are inserted into the table, they are stored based on the new table definition. As preexisting rows are updated, they are converted to the new table definition.

By storing table definitions, original and current, in the metadata and updating the metadata for a row "on-the-fly," dynamic schema changes are performed without impairing database performance and/or availability. Each backup taken after the first update following the first schema change can be used as the source for data recovery or to clone another DBMS with the latest table definitions because the tables are self-describing. Problems arise, however, if the source for data recovery is a backup taken prior to the first update following the first schema change. In this backup, the tables are not self-describing, i.e., the original and current table definitions not appended to the metadata. Moreover, the original table definition is not stored in the catalog after the table definition has been altered. Accordingly,

no record of the original table definition exists for this backup, and therefore, it cannot be used for data recovery.

Accordingly, a need exists for a system and method for preserving an original table schema in a database system that supports dynamic schema changes. The system and method should allow the database system to preserve a table definition before any alteration to the schema. The system and method should also allow any backup copy of the database, including one taken prior to a first update following a first schema change, to be used for data recovery. The present invention addresses such a need.

SUMMARY OF THE INVENTION

A method and system for preserving an original table schema for a table in a database system that supports dynamic table schema changes is disclosed. The method and system includes storing the original table schema for the table in a designated table prior to performing a schema change on the table.

By storing the original table schema in the designated table, a backup copy of the table that does not contain the original table definition can be used for data recovery. The original table schema can be removed from the designated table when all backups taken before the first update following the first table schema change become obsolete and/or unsuitable for data recovery.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a block diagram of a database system according to a preferred embodiment of the present invention.

Figure 2 is a flowchart illustrating a method for preserving an original table definition in a database system that supports dynamic schema changes according to a preferred embodiment of the present invention.

Figure 3 is a flowchart illustrating a method for rebuilding a table according to a preferred embodiment of the present invention.

DETAILED DESCRIPTION

The present invention relates to database management systems and more particularly to a method and system for preserving an original table schema in a database system that supports dynamic schema changes. The following description is presented to enable one of ordinary skill in the art to make and use the invention and is provided in the context of a patent application and its requirements. Various modifications to the preferred embodiment and the generic principles and features described herein will be readily apparent to those skilled in the art. Thus, the present invention is not intended to be limited to the embodiment shown but is to be accorded the widest scope consistent with the principles and features described herein.

According to the present invention, an original table definition, i.e. schema, is recorded in a table at or prior to a first schema change. The table is preferably stored in the database catalog along with other catalog tables. Once the first schema change is invoked, the dynamic schema change function converts all inserts or updates to the new format, i.e., self-describing rows, as described above. The existing rows that are not self-describing are defined by the original table definition stored in the table. Accordingly, each row, whether newly inserted, updated, or preexisting, is defined. Any backup copy of the database can now be used for data recovery.

To describe in more detail the present invention, please refer now to Figure 1, which is a block diagram of a database system according to a preferred embodiment of the present invention. As is shown, a plurality of clients 10 are coupled to a database 40 via a server 20. The server 20 includes a database management system (DBMS) 30, such as the DB2™ system developed by IBM of Armonk, New York. The DBMS 30 manages requests from a client 10 to access data stored in data tables 45 in the database 40, as well as the overall administration of the database 40. The DBMS 30 includes a dynamic schema change (DSC) mechanism 35 for supporting the dynamic schema alteration functionality.

Information related to the data is typically stored in a catalog 50 in the database 40. The catalog 50 typically includes catalog tables 55 describing the data tables 45 in the database system. Thus, definitions for data tables 45 are typically stored in a catalog table 55, as well as other information.

According to a preferred embodiment of the present invention, the catalog 50 includes an Original Version (OV) Table 100. The OV Table 100 is used to store a table definition immediately prior to a first schema alteration performed by the DSC mechanism 35. Although the OV Table 100 is preferably located in the catalog 50, those skilled in the art would readily appreciate that the OV Table 100 could be stored elsewhere in the database 40.

Figure 2 illustrates a flowchart illustrating a method for preserving an original table definition in a database system that supports dynamic schema changes according to a preferred embodiment of the present invention. The method begins by providing an Original Version (OV) Table 100 in step 202. As noted above, the OV Table 100 is preferably located in the catalog 50 along with other catalog tables (not shown). Next, in step 204, the original table schema for the data table 45a is stored in the OV Table 100 prior to a first schema change to

the data table 45a. Once the first schema change has been performed, the catalog table 55 is modified to reflect the new table schema.

From this point forward, any row inserted into the table 45a or any row updated will be defined by the new schema. Moreover, each row will be, or will be converted into, a self-describing row as described above, i.e., metadata describing the row's schema is appended. After a first table update, e.g., insert row or row update, the original table schema and new table schema are appended to the metadata. Thus, preexisting rows that are not updated will continue to be defined by the original table schema, found either in the metadata or in the OV Table 100.

During a data recovery process, the DBMS 30 can utilize any backup copy for a table 45a because each row is now defined by the metadata or by the original schema stored in the OV Table 100. Figure 3 is a flowchart illustrating a process for rebuilding a table 45a according to a preferred embodiment of the present invention. In step 402, a valid backup copy of the table is provided. The valid backup copy is an image of the table 45a that includes data that can be used to rebuild the table 45a, i.e., the data is not corrupted or otherwise unsuitable for data recovery. In step 404, the DBMS 30 refers to a first row in the backup copy, and in step 406 determines whether the row contains metadata. If metadata exists, then the table definition in the metadata is utilized to rebuild the row in step 408. If metadata does not exist, e.g., because the backup was taken prior to the first table update following the first schema change, then the table definition in the OV Table 100 is used in step 410. In step 412, a next row is accessed and steps 406 to 410 are repeated until the entire table 45a has been rebuilt. At the point when all rows in the table 45a have all necessary table definitions describing the data

rows, i.e., the metadata describes all the rows, the DBMS 30 will remove the original table schema for that table 45a from the OV Table 100.

Although the present invention has been described in accordance with the embodiment shown, one of ordinary skill in the art will readily recognize that there could be variations to the embodiment and those variations would be within the spirit and scope of the present invention. For example, while the preferred embodiment involves a DB2 system that supports dynamic schema changes, those skilled in the art would readily appreciate that the principles of the present invention could be utilized in a variety of database management systems. Accordingly, many modifications may be made by one of ordinary skill in the art without departing from the spirit and scope of the appended claims.